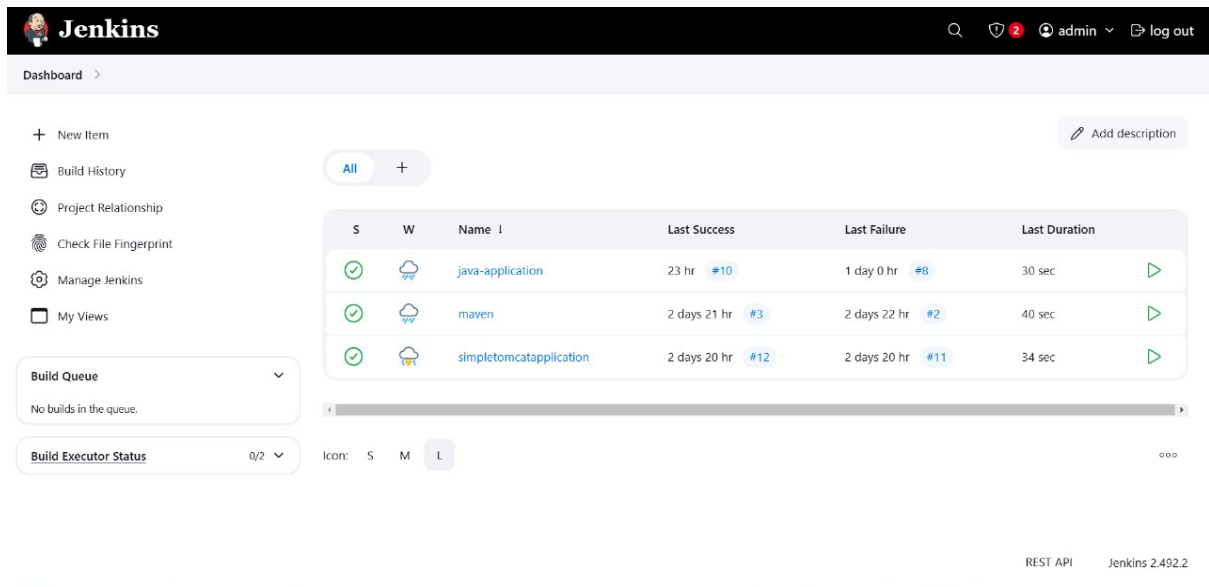


FINAL PROJECT

DAY-06

JAVA-APPLICATION DEPLOYMENT IN MINIKUBE

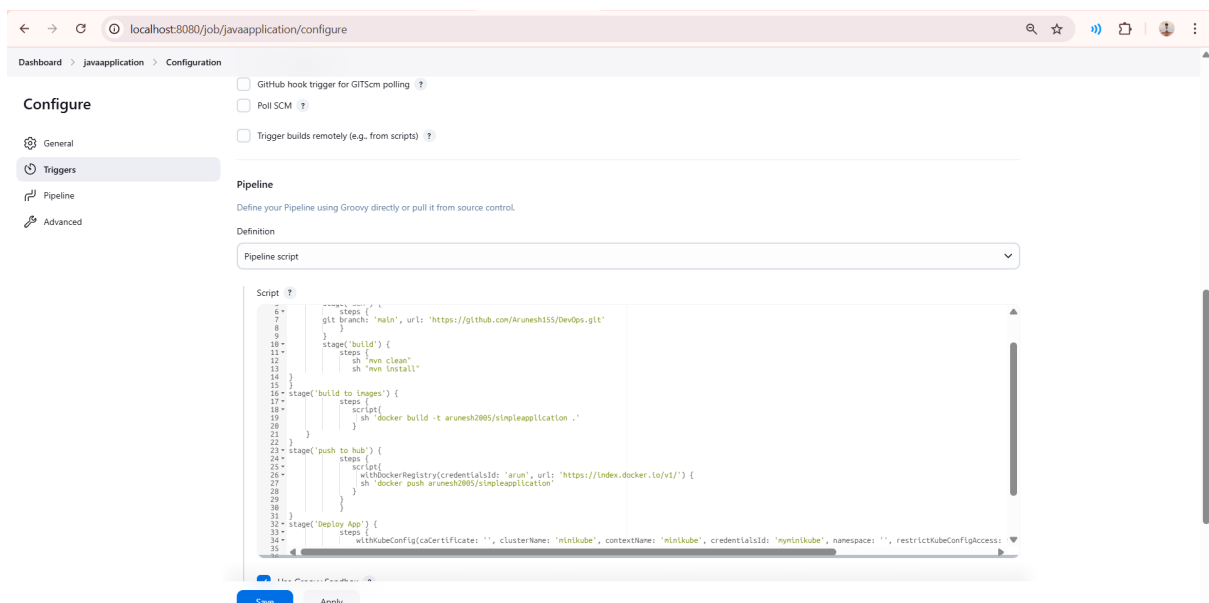
CREATION OF A PIPELINE PROJECT



The screenshot shows the Jenkins Dashboard. At the top, there's a header with the Jenkins logo, a search icon, a shield icon with a red '2', a user icon labeled 'admin', and a 'log out' button. Below the header, the 'Dashboard' link is active. On the left sidebar, there are links for 'New Item', 'Build History', 'Project Relationship', 'Check File Fingerprint', 'Manage Jenkins', and 'My Views'. The main area displays a table of builds. The table has columns for 'S' (status), 'W' (weather icon), 'Name', 'Last Success', 'Last Failure', and 'Last Duration'. There are three rows of builds: 'java-application' (status: success, duration: 30 sec), 'maven' (status: success, duration: 40 sec), and 'simpletomcatapplication' (status: success, duration: 34 sec). Below the table, there's a 'Build Queue' section showing 'No builds in the queue.' and a 'Build Executor Status' section showing '0/2'.

S	W	Name	Last Success	Last Failure	Last Duration
✓	☁	java-application	23 hr #10	1 day 0 hr #8	30 sec
✓	☁	maven	2 days 21 hr #3	2 days 22 hr #2	40 sec
✓	☁	simpletomcatapplication	2 days 20 hr #12	2 days 20 hr #11	34 sec

PIPELINE SCRIPT



The screenshot shows the Jenkins 'Configure' page for a pipeline named 'javaapplication'. The 'Pipeline' tab is selected. The 'Definition' section shows a 'Pipeline script' dropdown. The script is as follows:

```
6+  pipeline {
7+    agent {
8+      git {
9+        branch 'main', url: 'https://github.com/Arumesh155/DevOps.git'
10+      }
11+    }
12+    stage('build') {
13+      steps {
14+        sh 'mvn clean'
15+        sh 'mvn install'
16+      }
17+    }
18+    stage('build to images') {
19+      steps {
20+        script {
21+          sh 'docker build -t arumesh2005/simpleapplication .'
22+        }
23+      }
24+    }
25+    stage('push to hub') {
26+      steps {
27+        withDockerRegistry(credentialId: 'arun', url: 'https://index.docker.io/v1/') {
28+          sh 'docker push arumesh2005/simpleapplication'
29+        }
30+      }
31+    }
32+    stage('Deploy App') {
33+      steps {
34+        withKubeConfig(caCertificate: '', clusterName: 'minikube', contextName: 'minikube', credentialId: 'myminikube', namespace: '', restrictKubeConfigAccess: '') {
35+          sh 'kubectl apply -f deployment.yaml'
36+        }
37+      }
38+    }
39+  }
```

PIPELINE SCRIPT CODE

```
pipeline {
    agent any

    stages {
        stage('scm') {
            steps {
                git branch: 'main', url: 'https://github.com/Arunesh155/DevOps.git'
            }
        }
        stage('build') {
            steps {
                sh "mvn clean"
                sh "mvn install"
            }
        }
        stage('build to images') {
            steps {
                script{
                    sh 'docker build -t arunesh2005/simpleapplication .'
                }
            }
        }
        stage('push to hub') {
            steps {
                script{
                    withDockerRegistry(credentialsId: 'arun', url: 'https://index.docker.io/v1/') {
                        sh 'docker push arunesh2005/simpleapplication'
                    }
                }
            }
        }
    }
}
```

```

    }
  }
}
stage('Deploy App') {
  steps {
    withKubeConfig(caCertificate: '', clusterName: 'minikube', contextName: 'minikube',
credentialsId: 'myminikube', namespace: '', restrictKubeConfigAccess: false, serverUrl:
'https://192.168.39.226:8443'){

      sh 'kubectl apply -f deploy.yaml --validate=false'
    }
  }
}
}
}
}
}

```

DEPLOYMENT FILE

DEPLOY.YAML

```

apiVersion: apps/v1
kind: Deployment
metadata:
  name: my-deploy
  labels:
    name: my-deploy
spec:
  replicas: 5
  selector:
    matchLabels:
      apptype: web-backend

```

```
strategy:
  type: RollingUpdate
template:
  metadata:
    labels:
      apptype: web-backend
  spec:
    containers:
      - name: my-app
        image: arunesh2005/simpleapplication
        ports:
          - containerPort: 9000
```

```
apiVersion: v1
kind: Service
metadata:
  name: my-service8
  labels:
    app: my-service
spec:
  type: NodePort
  ports:
    - port: 9000
      targetPort: 8080
      nodePort: 30002
  selector:
    apptype: web-backend
```

CRENADENTIALS

Dashboard > Manage Jenkins > Credentials > System > Global credentials (unrestricted) > config

Update
Delete
Move

config

Usage

This credential has been recorded as used in the following places:
Note: usage tracking requires the cooperation of plugins and consequently may not track every use.

[java:application](#) ⊖ #7 ⊕ #8

localhost:8080/job/java-application/

REST API Jenkins 2.452.2

STAGE VIEW OF BUILD

localhost:8080/job/javaapplication/

Jenkins

Dashboard > javaapplication

Status 🟢 javaapplication Add description

Changes
Build Now
Configure
Delete Pipeline
Full Stage View
Stages
Rename
Pipeline Syntax

Stage View

	scan	build	build to images	push to hub
Average stage times: (Full run time: ~29s)	2s	4s	1s	15s
1428 Test changed	1s	4s	378ms	21s
1429 Test changed	2s	1s	542ms	1min 9s
1541 Test changed	1s	3s	188ms	23s
1542 Test changed	4s	3s	178ms	218ms
1543 Test changed	3s	4s	688ms	143ms
1547 Test changed	3s	3s	410ms	218ms

Builds

Filter

Today

- 🟢 #7 8:00 AM
- 🔴 #6 8:00 AM
- 🟢 #5 8:11 AM
- 🔴 #4 8:10 AM
- 🔴 #3 8:00 AM
- 🔴 #2 8:07 AM
- 🔴 #1 8:00 AM

localhost:8080/job/javaapplication/?

Jenkins

Dashboard > javaapplication > #5

Status 🟢 #5 (Mar 21, 2025, 8:11:11AM) Add description Keep this build forever

Changes
Console Output
Edit Build Information
Delete build

Timings

Git Build Data

Pipeline Overview

Pipeline Console

Restart from Stage

Replay

Pipeline Steps

Workspaces

Previous Build

Started by user [admin](#)

This run spent:

- 8 ms waiting
- 29 sec build duration
- 29 sec total from scheduled to completion.

Revision: 6562f38107722f1d40595a1bd4f66da9770ec90c
Repository: <https://github.com/krusenit155/DevOps.git>

- refs/remotes/origin/main

No changes

Started 41 sec ago
took 29 sec

8°C Sunny

Search

13:42 21-03-2025

```
Dashboard > javaapplication > #5

Login Succeeded
[Pipeline] {
[Pipeline] sh
+ docker push arunesh2005/simpleapplication
Using default tag: latest
The push refers to repository [docker.io/arunesh2005/simpleapplication]
2fc088b0b11: Preparing
62ff1acae04: Preparing
34f7184834b2: Preparing
5836ece09bdf: Preparing
72e830a4dff5: Preparing
62ff1acae04: Layer already exists
34f7184834b2: Layer already exists
72e830a4dff5: Layer already exists
5836ece09bdf: Layer already exists
2fc088b0b11: Pushed
latest: digest: sha256:354b3e595b04c5947704608757d1521bba7c1b0b0e2df236a4824430828f9de2 size: 1305
[Pipeline] }
[Pipeline] // withDockerRegistry
[Pipeline] }
[Pipeline] // script
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```

```
arunesh@Arunesh:~$ kubectl get pods
NAME                READY   STATUS    RESTARTS   AGE
my-deploy-855fc965b7-z6h7n  0/1     ContainerCreating  0          3m8s
arunesh@Arunesh:~$ kubectl get pods
NAME                READY   STATUS    RESTARTS   AGE
my-deploy-855fc965b7-z6h7n  1/1     Running    0          4m16s
arunesh@Arunesh:~$ kubectl apply -f new_dep.yml
deployment.apps/my-deploy configured
service/my-service unchanged
arunesh@Arunesh:~$ minikube service my-service
-----
| NAMESPACE | NAME   | TARGET PORT | URL                               |
|-----|-----|-----|-----|
| default   | my-service | 8080        | http://192.168.49.2:30078        |
-----
* Starting tunnel for service my-service.
-----
| NAMESPACE | NAME   | TARGET PORT | URL                               |
|-----|-----|-----|-----|
| default   | my-service |             | http://127.0.0.1:36897           |
-----
🔗 Opening service default/my-service in default browser...
🔗 http://127.0.0.1:36897
! Because you are using a Docker driver on linux, the terminal needs to be open to run it.
http://192.168.49.2:30078http://192.168.49.2:30078* Stopping tunnel for service my-service.
arunesh@Arunesh:~$ ^C
arunesh@Arunesh:~$ ^C
arunesh@Arunesh:~$ curl http://192.168.49.2:30078/
<doctype html><html lang=en"><head><title>HTTP Status 404 - Not Found</title><style type="text/css">body {font-family:Tahoma,Arial,sans-serif;} h1, h2, h3
, b {color:white;background-color:#525D76;} h1 {font-size:22px;} h2 {font-size:16px;} h3 {font-size:14px;} p {font-size:12px;} a {color:black;} .line {height:1px;background-color:#525D76;border:none;}</style></head><body><h1>HTTP Status 404 - Not Found</h1><hr class="line" /><p><b>Status Report</b></p><p><b>Description</b> The origin server did not find a current representation for the target resource or is not willing to disclose that one exists.</p><hr class="line" /><h3>Apache Tomcat/9.0.102</h3></body></html>
arunesh@Arunesh:~$ curl http://192.168.49.2:30078/my-app/
<html>
<body>
<h2>Hello World!</h2>
</body>
</html>
```

